GUIDELINES FOR AUTHOR-PREPARED INPUT FOR TYPESETTING

by
Barbara Beeton, Technical Assistant
Information Services
American Mathematical Society
Providence, Rhode Island

Summary. Electronic input for typesetting requires more care and consistency than paper manuscripts or even than word processing input. This is especially true when the author does not know (or wish to learn) the details of the typesetting system ultimately to be used. Even so, by rigorously identifying each different logical element of a document, an author can prepare an electronic manuscript that can successfully be transformed into composed pages for publication with minimal post-processing.

Background

The American Mathematical Society (AMS) is now using two composition languages for production of journals and books. Both were designed especially for mathematics, so in certain respects they are not typical. Nonetheless, mathematical documents have physical structures similar to most other scholarly works, and any competent typesetting program must deal with the common problems.

A language created by Science Typographers, Inc., has been in use at AMS since 1975; this we refer to as the *STI language*[1]. It uses very short codes to represent symbols and typesetting commands:

```
*gG, *ga
```
Γ, α

```
03g *ix
```
$\log x$

```
$($z*bs$si=1$sn$t*gF(x)*di$)
```
$$\sum_{i=1}^{n} \Phi(x)_i$$

```
$($fx*u2*b-y*u2$s2$t=z*u2$)
```
$$\frac{x^2 - y^2}{2} = z^2$$

The STI language is very compact and efficient for production use by skilled keyboarders, but it is not attractive to authors or secretaries for preparing occasional manuscripts.

The other language used at AMS, since 1979, is TEX[†][3,4], "tau epsilon chi", from the Greek stem for art and technology. TEX was created by a computer scientist, Donald Knuth of Stanford University, in order to compose his own books. His series, *The Art of Computer Programming*, is planned to reach seven volumes; at an average of five years per volume (three volumes so far), it became clear that the typesetting method used for Volume 1 (Monotype) would not be available for Volume 7. A desire for both consistency and beauty was the impetus for creating TEX. The TEX language uses mnemonic codes to represent symbols and typesetting instructions:

```
$\Gamma, \alpha$
```
Γ, α

```
$\log x$
```
$\log x$

```
$$\sum_{i=1}^n \Phi(x)_i$$
```
$$\sum_{i=1}^{n} \Phi(x)_i$$

```
$${x^2 - y^2 \over 2}= z^2$$
```
$$\frac{x^2 - y^2}{2} = z^2$$

TEX is more accessible to authors than the STI language, but it is not yet proven as a production tool, being both more verbose and still "under construction".

More important than the differences between these composition languages is the fact that each requires rigorous delineation of all symbols and structural elements, to permit accurate transformation of the input to typeset copy in the desired form. Both languages permit either explicit typesetting specifications (12-point boldface italic) or generic identification (title). For author-prepared input, only generic identifiers should be used, to permit unambiguous application of the journal or book series style rules.

Deciding whether or not to use author-prepared input

There are two principal reasons for using author-prepared input: cost savings, and author control over his work. Cost is by far more important to the publisher. Although it would seem that capturing the original keystrokes for a manuscript should result in lowered processing costs, several areas should be examined closely before deciding to accept an author-prepared input file:

- Does the document contain a large proportion of complex material—figures, tables and display mathematics?
- Is the document short (less than book length)?
- Is the author unfamiliar with the composition system to be used by the publisher?

If the answer to any of these questions is yes, it may not be worth the effort to try to reuse the original keystrokes. A skilled keyboarder can rekey complex material more quickly and accurately than he can insert typesetting codes into an existing file, while the cost of a programmer to write a one-time translation algorithm is likely to exceed that of any keyboarding effort.

If the author can use the same composition software as the publisher, compatibility problems will be minimized, and the cost picture brighter. This might be the case if the author is located at a university with an academic computer

[†] TEX is a trademark of the American Mathematical Society.

If no paper accompanies an electronic manuscript, you can let the compositor experiment, but remember — author-prepared input is supposed to save time and money.

Assuming no insoluble problems arise, the compositor will return several things to the editor:

- Proof copy.
- Any documents the editor submitted with the job.
- A list of systematic errors and other problems found in the file.

All such problems should be communicated to the author, along with a statement of what has been done to fix them. It's also useful to accumulate this information, adding it to your list of "hints for authors".

If time and policy permit, you might allow the author one last look at his paper before it's sent to press. He should be encouraged not to make changes at this stage, but if he does, they should be put in the file by the compositor; the author should not send a new file.

Guidelines for the compositor

The compositor's most important task is to learn everything possible about the composition system and other computer tools available, in order to be able to extract proof and camera copy from the system. By going about the job in a systematic way, it is possible to minimize the time and effort involved, and turn out a quality result.

Even before introducing a file to the composition program, take a few steps to help avoid problems:

- Get the list of codes provided by the author.
- Check the input notation; if it isn't standard, change it.

A good text editor will help here. If the author didn't use the same composition system, changes may be extensive. If he didn't adhere to the author guidelines or made other systematic errors, make a list of such problems to be returned to him. A particular problem that might arise if TEX is the common language, is the redefinition of "basic" commands; this practice is dangerous in a journal environment, where papers from different sources are combined, and the author should be specifically warned against it.

- Work out code translations on paper before making them on the file.

This includes determining the order in which changes should be done, to avoid introducing ambiguities.

- Before changing the file, keep a backup copy.

This is cheap insurance if something goes wrong, and a machine comparison of the "before" and "after" versions is often the quickest way of verifying that changes are complete and correct.

- Use the author's paper copy of the file for reference.

Notes are made more easily on paper than in a computer file. Also, if something looks strange in the file, or you have trouble loading it onto your computer, it may be clear on the paper copy.

- If a problem seems insoluble, stop making changes, but do scan the rest of the file to try to identify other problems that may require the author's help.
- Keep a record of problems and their solutions.

Keep a copy with the job, to be returned to the author, and add new ones to your cumulative list — the same problems may occur again in other manuscripts.

Finally, proof will emerge from your output device. Give it to the editor with your analysis of problems to be reported to the author. And get ready for the next job.

Conclusion

Having described some ways to handle author-prepared electronic manuscripts, I must report, sadly, that the AMS is not yet in a position to practice this on any grand scale. We've received only a few papers prepared using TEX, and are not actively soliciting more at the moment. Nor do we contemplate accepting any input files intended for processing via the STI system. We are installing a new version of TEX, TEX82, which is just beginning to be in general distribution; we hope this will help us fulfill the promise we saw when we began to experiment with TEX four years ago.

Many of these remarks have been cautionary, rather than an invitation to start gathering manuscripts on magnetic media. But I don't want to discourage anyone from trying out this new technology.

Composition of scholarly journals with the aid of computers has been happening for at least a decade. At the AMS, we've been preparing input for our primary journals in-house since 1975. We've found ourselves being pioneers when we expected to take a program down off a shelf, feed it into the computer and have beautifully composed pages appear automatically. We've learned to wring every last feature out of these programs, and when we found them wanting, have gone back to their creators and asked for improvements. The STI system now churns out tens of thousands of feet of copy a year, although the knife is still required to divide the copy into pages. TEX is just beginning to take its place as a production tool, delivering documents for internal use as well as finished journal pages, entirely in-house.

Typesetting capability has moved from the composition house into the publisher's office, and it's now coming within the reach of authors. A new generation of authors is growing up with computers — a keyboard may become more familiar than a pencil to many of them. They'll be eager to send you their words on electronic files, and if you're not ready for them, what might have been a fruitful venture could become instead a colossal bother. The new technology really should make it quicker and easier for authors to get their ideas into print. One can only hope that their literary abilities keep pace with their technical competence.

References

[1] *Reference Manual for Copy Editors, Production Assistants, and Keyboarders in Computer-Assisted Photocomposition.* Sixth edition. Science Typographers, Inc., Medford, New York, 1982.

[2] *Word Processing for Typesetting: How to Prepare Manuscripts in Machine-Readable Form for Publication.* Science Typographers, Inc., Medford, New York, 1982.

[3] Donald E. Knuth, *TEX and METAFONT, New Directions in Typesetting.* Digital Press/American Mathematical Society, Bedford, Massachusetts, 1979.

[4] ———, *The TEXbook.* Addison-Wesley Publishing Company, Reading, Massachusetts, to appear.

center or computer science department — the composition languages TEX and TROFF/EQN (components of UNIX†) are becoming increasingly popular and available in such environments, and manuals on how to use them can be obtained at many college bookstores. It's much less likely for a proprietary typesetting language (such as the STI language) to be available on an academic computer.

Once the decision is made to accept author-prepared input, everyone involved must find out what is expected of him, be prepared to do it, and not hesitate to ask questions should any problem arise. Success requires the cooperation of three key participants:

- The *author* (and usually his secretary as well).
- The *production editor*, responsible for enforcing style, and the bridge between author and compositor.
- The *compositor*, responsible for technical details and interacting with the composition software to produce publishable copy.

Guidelines for the author

Before starting to create a file, the author should obtain from the publisher any available instructions on input format and acceptable forms of magnetic media. He then must make sure that his computer file embodies his intentions precisely. Whatever reaches the publisher on magnetic tape or floppy disk is what will get published; the author may not get another chance to fix it up.

Some general guidelines hold regardless of input language to be used. The following suggestions are based on a booklet prepared by the STI staff[2].

- Be consistent. Always do the same thing in *exactly* the same way.
- Be precise. Never do two *different* things the same way.

Two different things that may look the same in print are the first reference to a *technical term*, and a word or phrase given *emphasis*. Technical terms are frequent candidates for index entries, and by tagging them uniquely the author can increase the usefulness of a file.

- Distinguish between look-alikes.

Some commonly confused symbols:

l (ell) and 1 (one); O (oh) and 0 (zero);
- (hyphen), – (en-dash), — (em-dash) and − (minus).

- Use visible codes to give formatting instructions.

Spaces, tabs and carriage returns can't be distinguished on a screen or a printout. Besides, typesetting programs use other methods to center headings and break lines. Identify the type of data by tagging it.

- Use codes to identify

  Symbols or characters not on your keyboard.
  Symbols reserved for "escape characters".
  Data elements — title, theorem, etc.
  Formatting instructions — paragraph, space for figures, etc.

Most keyboards have only a limited number of characters. One or more will have to be reserved as "escape characters", to indicate that what follows, or comes between a pair, isn't text, but a special instruction. The STI system uses the dollar sign, at sign and asterisk; TEX uses the backslash, dollar sign and braces. A more neutral scheme might bracket instructions between greater- and less-than signs: `<title>...<endttl>`. Whatever coding scheme is used, apply it precisely and completely.

- Indentify both beginning and end of data elements.

It's easier for a compositor, and *much* easier for a computer to ignore redundant instructions than to decide where to insert missing ones.

- Distinguish between levels of headings and sub-headings.

Headings define the structure of a document, and can be thought of as its outline. Clear tagging of important structural distinctions frees the document from dependence on a single composition system.

When your "manuscript" is complete, create a file in magnetic form. Send it to the editor along with a paper copy — this will be invaluable if there's any trouble reading the file. If you've used your own notation, and especially if your notation isn't explained in a readily available source (such as the TEX manual, or a manual you got from the publisher before you began), send a complete list of the codes used and what they're supposed to represent. Even if the notation is standard, a list will tell the editor what special symbols and features the manuscript will need.

When proof is ready, the editor may send you a copy, along with comments from the compositor. Take note of them, so your next project will go even more smoothly.

Guidelines for the editor

The editor is the bridge between author and compositor, and will get questions from both sides.

- Know the composition system, what it can and can't do.

If an author's questions can be answered before he puts the manuscript on magnetic tape, they can be kept from becoming editorial problems. If the author requests notation that isn't available, ask him to use something else. But try to accommodate reasonable requests, and be ready to explain why some requests aren't reasonable.

- Expect all manuscripts to require some adjustments.

Even if the author is an expert in the same system you're using, and has matched your style perfectly, such items as logos, copyright notice and editorial receipt date will have to be added, along with correct page numbers. The trick is to keep other changes to a minimum.

On the other side of the editor is the compositor. He should be expected to be an expert on the composition system. In fact, if technical questions arise, it's often better to ask the compositor than to look up the answers in a manual (assuming that composition is performed in-house, not several thousand miles away). But the compositor also expects guidance from the editor.

- Give the compositor all relevant documents from the author, along with the magnetic file.

The list of notation is most important. If the author has provided a typeset preprint, mark on it the things that need attention — style changes, journal boiler plate, and the like. If only a typescript was delivered, tell the compositor what journal the paper is assigned to, and trust his technical knowledge to do what is needed to produce proof.

- If information from the author is inadequate, go back to the author before giving anything to the compositor.

† UNIX is a trademark of Bell Laboratories.